Claim Listing

1. (Previously presented) An asymmetric data processing system comprising:

a first group of one or more host computers, each comprising a memory, a network interface and one or more Central Processing Units (CPUs), each host computer accepting and responding to requests to process data;

a second group of two or more Job Processing Units (JPUs), operating autonomously and asynchronously from one another, each JPU comprising a memory, a network interface, a data interface with exclusive access to one or more sources of data, and one or more general purpose CPUs, each JPU in the second group being responsive to requests received from a host computer to execute jobs and return results independent of execution at other JPUs, the jobs containing instructions for the processing of a particular subset of data under the JPU's exclusive control; and

a network connecting the network interfaces within each group and between the two groups.

2. (Original) The system of claim 1 wherein the data comprises structured records.

3. (Original) The system of claim 1 wherein the data comprises a mixture of fixed and variable length fields of various data types.

4. (Original) The system of claim 3 wherein the data further comprise header information describing fields with null values and offsets of variable length fields.

5. (Original) The system of claim 1 wherein the sources of data comprise one or more storage devices which are directly accessed by no other JPU in the second group and by none of the host computers in first group.

6. (Original) The system of claim 1 wherein the sources of data comprise an external source of streaming data, such that the streaming data is directly accessed by no other JPU in the second group and by none of the host computers in the first group.

7. (Original) The system of claim 1 wherein autonomous operation is such that host computers in the first group do not coordinate processing across JPUs.

8. (Original) The system of claim 1 wherein asynchronous operation is such that host computers in the first group operate in synchronism with the JPUs.

9. (Original) The system of claim 5 in which JPUs in the second group manage the storage devices autonomously, such that they have exclusive responsibility for the mapping between the location and representation of data in memory and the location and representation of data within the storage devices.

10. (Original) The system of claim 9 in which JPUs in the second group manage their associated local storage devices by performing at least one function selected from a group consisting of: storage allocation and deallocation; insertion, deletion and retrieval of records; creation and deletion and maintenance of tables, views and indices; mirroring and replication; and compression and decompression.

11. (Original) The system of claim 10 in which the JPUs in the second group further comprise a storage manager component which is responsible for hiding details of storage management from other components of the JPUs.

12. (Original) The system of claim 11 in which the storage manager component checks requests to insert record data into a table to ensure that the record data conforms to the table's definition.

13. (Previously presented) The system of claim 1 in which the JPUs in the second group manage transactions autonomously, containing software which is responsible for at least

one of the following functions: starting, pre-committing, committing and aborting transactions against data on the JPU.

14. (Original) A system as in claim 1 wherein the JPUs in the second group control concurrent access to data that is local to the JPU, containing software which is responsible for locking the local data and identifying dependencies between transactions that process local data.

15. (Original) System of claim 1 in which the JPUs in the second group perform mirroring autonomously, by ensuring that modifications to data local to a first JPU are replicated redundantly on another device.

16. (Original) System of claim 15 in which the device containing replicated data of the JPU is a second JPU.

17. (Original) The system of claim 1 in which the JPUs in the second group may receive new jobs before completing older jobs, and where the resources required to satisfy jobs are scheduled locally and autonomously by the JPUs that own the resources.

18. (Original) The system of claim 1 wherein the JPU's report job results to at least one host.

19. (Original) The system of claim 18 wherein jobs are assigned a job identifier and wherein the host additionally coordinates completion of jobs by determining when all active JPU's report results for a particular job.

20. (Original) The system of claim 18 wherein the JPU's further provide a JPU identifier with results data, so that job completion can be coordinated by all active JPU's.

21. (Original) The system of claim 20 wherein coordination of results is performed by a Large Job Processing Unit (LJPU) in a third group of processors .

22. (Original) The system of claim 1 wherein a host in the first group sends a job to a plurality of JPUs in the second group by broadcasting a message containing the job onto the network, without specifying the identity of any JPUs to receive the job.

23. (Original) The system of claim 22 wherein the JPUs that receive the same jobs from one or more hosts may process those jobs in different orders and at different times.

24. (Original) The system of claim 1 wherein each JPU in the second group further comprises a scheduling component, and each JPU processes its assigned jobs and returns results to a requesting host in the order and at the time that the scheduling component specifies.

25. (Original) The system of claim 1 in which a host that issues a request to multiple JPUs accepts replies from the JPUs when provided under control of the JPU operating autonomously, and not in any order specified by the host.

26. (Original) The system of claim 1 in which at least one JPU has an operating system capable of receiving and processing multiple jobs at a given time.

27. (Original) The system of claim 26 wherein the operating system supports overlapping job execution.

28. (Original) The system of claim 1 in which JPUs in the second tier are embedded components that are not directly accessible to applications that present data processing requests to the hosts.

29. (Original) The system of claim 28 in which the hosts in the first group are exclusively responsible for interfacing to external applications, thereby supporting the use of JPUs having different processing capabilities, without requiring changes to be made to the applications making requests.

30. (Original) The system of claim 28 in which execution of requests made by a first application cannot affect the correct execution of requests made by other applications.

31. (Original) The system of claim 28 in which a pre-existing application that makes a request in a standard query language of the system, results in the host distributing jobs to one or more JPUs in the second group, without having to change the pre-existing application.

32. (Original) The system of claim 1 in which the data processed by the host computers in the first group is partitioned into two or more subsets, such that the processing of data in each subset is the primary responsibility of no more than one JPU. .

33. (Original) The system of claim 32 in which the identity of a JPU primarily responsible for processing a given subset of data is determinable as a function of the data.

34. (Original) The system of claim 1 additionally comprising:

a third group of Large Job Processing Units (LJPUs), each LJPU being responsive to jobs, the LJPUs having greater memory and processing capabilities than the JPUs; and

network also connects LJPUs in the third group to the computers of the other groups.

35. (Original) The system of claim 34 wherein LJPUs share memory resources.

36. (Original) The system of claim 33 wherein the LJPUs have an execution engine responsible for coordinating results from the JPUs.

37. (Original) The system of claim 1 wherein the computers in the first group are arranged in a symmetric multiprocessing architecture.

38. (Original) The system of claim 1 wherein the computers in the second group are arranged in a massively parallel architecture.